Indoor Location Systems for Pervasive Computing

Joshua A. Tauber
`josh@lcs.mit.edu`

8 August 2002

Abstract

As computing devices have been shrinking and proliferating and networking technologies have enabled ever greater connectivity, interest in the category of location-aware applications has grown. Determining location in cluttered, indoor environments presents special challenges.

This paper identifies common techniques and technologies for estimating the location of pervasive computing elements indoors. We then examine a set of properties by which to evaluate location systems, apply this evaluation method to survey a number of existing systems, and discuss the trade-offs among them. We focus particularly on the Ad Hoc Location System (AHLoS), RADAR, Cricket, Cricket Compass, and Active Bats. Finally, we identify research opportunities in sensor fusion, software support for localization, distributed algorithms of localization, analysis of beaconing protocols, and bootstrapping system designs.

1 Introduction

Over the entire history of the field, computing devices have been shrinking and proliferating. Furthermore, over the last three decades, networking technologies have been keeping pace to enable ever greater connectivity of every smaller — and now — mobile devices. As we approach the level of ubiquitous network connectivity and pervasive, mobile devices, the enticing new category of context-aware applications has been proposed. One important dimension of context is location. Location-aware applications include navigational guides, location-specific enhanced-reality annotations, and "follow-me" services.

Another emerging application area of pervasive computing are ad hoc sensor networks. Ad hoc networks are distinguished from traditional mobile computing settings in that computation nodes are assumed to be small, inexpensive, homogeneous, cooperative, and often relatively autonomous. A number of location-aware protocols have been proposed for ad hoc routing and networking.

Determining location in indoor environments present special engineering, social, and regulatory challenges. The scale of of indoor environments is such that the location information desired is fairly fine grained. The environment is cluttered, making sensing difficult. Privacy concerns arise from the specter of "big brother" systems that follow users around. Finally, health concerns and commercial regulation limit the available technologies.

This paper defines indoor location systems and then identifies common techniques and technologies for estimating the location of pervasive computing elements indoors. We then examine a set of properties by which to evaluate location systems. We apply this evaluation method to survey a number of existing systems including the ad hoc location system (AHLoS), RADAR, Cricket, Cricket Compass, and Active Bats. We discuss the trade-offs among these systems. Finally we identify research opportunities in the area of indoor location systems for pervasive computing.

2 Techniques and Technologies

Fundamentally, location systems associate elements of a symbolic system with the positions of people and objects in the physical world. That is, these systems associate symbolic labels with physical entities. Locations systems use sensor readings to measure the physical properties of the system and then perform some combining algorithmic steps to compute locations. Below we discuss the combining techniques and note the which properties of the sensors are relevant to the techniques. We then discuss the common sensor technologies in terms of these properties.

2.1 Techniques

To bootstrap a location system, some nonempty set of locations are assigned symbolic labels *a priori*. Thereafter, the location of any other point is computed with respect to these *reference points*. The process is called *positioning* [Van01]. Positioning occurs in two steps. First sensor measurements are obtained, then the measurements are combined to deduce the location of the unknown point(s). Generally, measurement involves the transmission and reception of signals between elements of the system.

Two location systems are *independent* if we cannot map locations in the one system onto the symbology of the other system. Systems are *convertible* if such a map does exist. While no absolute location symbology exists, many government organizations maintain a number of different systems with global-coverage (*e.g.*, Universal Transverse Mercator coordinates) that are essentially convertible among themselves.[1] We say a system is *geographic* if it is convertible to one of these global systems.

2.2 Measurement

2.2.1 Measuring distance

The metric most often used to assign labels to objects is the distance of the object from some set of other objects whose symbolic labels are already known (either by previous measurement and calculation, or *a priori*). Distance is estimated by sensing the characteristics of signals from elements of the location system.

Two approaches are commonly used to estimate distance using emitted signals. The first approach measures the attenuation of signal strength at a receiver. Attenuation-based methods attempt to calculate the signal loss due to propagation. Theoretical and empirical models are used to translate the difference between the transmitted signal strength and the received signal strength into a range estimate.

The second approach measures the time of flight (ToF) of a signal. If the signal propagation speed is known (or can be calculated), signal ToF trivially translates to distance. The key difficulty is to measure transmit and receive times on the same time scale. The precision with which times need to be synchronized is proportional to the speed of the signal and the desired precision of the distance measurement. For example, a synchronization error of one microsecond when timing an electro-magnetic signal (in a vacuum) results in a distance estimation error of one microlightsecond (1μlightsecond $\sim$ 300m).

Precisely synchronizing clocks of mobile units in a location system adds complexity, expense, and often weight (for extra energy storage) to the mobile units. Therefore, most systems do not require mobile elements of a system to synchronize. Four standard workarounds are common:

Single source; multiple, located, synchronized receivers A signal is broadcast from one location and received at several known locations with synchronized receivers. The receivers share arrival times, compute the *time difference of arrival* (TDoA), and solve for the time of flight. Typically, the receivers are synchronized through a wired network connection.

Multiple, located, synchronized sources; single receiver Multiple signals are broadcast from synchronized elements at distinct locations. The receiver measures the TDoA and solves for time of flight.

Single source; multiple, synchronized signals Two signals with different propagation speeds are broadcast simultaneously from a single source. Each receiver can use the ratio of the known propagation speed and the time difference of arrival to compute the time of flight. For typical indoor distances, if the faster signal is electro-magnetic and the other much slower (*e.g.*, sound), the ToF of the fast signal can be neglected and the TDoA is a sufficient approximation of the ToF of the slow signal.

Round-trip time One element of the system acts as a signal “mirror” rebroadcasting any received signals, possibly remodulating the signal to add information or change channels. The signal’s originator uses a single clock to measure the round trip ToF by subtracting the (known) fixed delay at the mirror and dividing by two.

---

[1] There are in fact, a *huge* number of government organizations concerned with location standards including U.S. Geological Survey, National Geodetic Survey, National Imagery and Mapping Agency, National Institute of Standards and Technology, and National Oceanic & Atmospheric Administration.

2.2.2 Measuring angles

Angle of arrival (AoA) of a signal against some arbitrary baseline can also be measured using either signal strength or time difference of arrival. Signal-strength-based systems compare the received SS across a spectrum of angles and select the angle of maximum strength as the receive angle. Time-based methods use arrangements of receivers to measure the TDoA and, thus the difference in distance from each receiver to the transmitter. When combined with the known arrangements of the receiver array, this differential distance information is sufficient to solve for the angle of arrival.

2.3 Combining

Location systems combine basic measurements such as distance, angle, or temperature to compute to assign locations. Four methods are common:

Multilateration Multilateration is the computation of location using measured distances from reference points. The two dimensional location of a point can be computed from the distances of that point to three non-collinear, reference points. In three dimensions, four non-coplanar constraining points are required. Additional measurements can be used to solve for additional unknowns. For example, measuring the distance to a single additional point allows location computation even if only distance ratios are known (*e.g.*, if signal propagation speed ToF-based measurements). Ranges to additional reference points can also be used to reduce error by finding a best fit in an over-constrained system.

Angulation Angulation is the computation of location using measured angles from reference points It is simply the angle-based analog of multilateration. Together, the two methods are called *triangulation*.

Proximity An alternative to explicit computation of location is to approximate the position of an element by assigning it the same label as that of a proximate reference point. The "closest" (by some metric) reference point to the unknown point is selected as the location of the unknown point. Common metrics include statistical functions of distance and physical contact.

Scene analysis Scene analysis is a catch-all phrase used to group a diverse set of holistic or complex methods. These methods recognize features of the environment that may not be so easily categorized as distance or angle. Example features include edge or motion detection in video images and received radio signal characteristics such as multipath or signal strength patterns.

2.4 Technologies

2.4.1 Signals

The relevant characteristics of signal propagation include the range, propagation speed, available bandwidth, diffraction and reflection characteristics, regulatory constraints, interference, power constraints, safety, and cost.

Infrared Due to their ubiquitous deployment infrared (IR) transceivers are inexpensive, compact, and low power. IR propagation is fast but effective bandwidth is limited by interference from ambient light and from other IR devices in the environment. IR signals reflect off most interior surfaces but diffracts around few. Typical range is up to 5 meters.

Radio-frequency Radio-frequency (RF) signals offer several benefits over IR. RF signals diffract around and pass through common building materials. RF signals compare favorably to IR in propagation speed, bandwidth, and cost. Since the RF spectrum is heavily regulated, typical systems operate at 900MHz or 2.45GHz and comply with Part 15 FCC regulations so as not to require licensing. Transmission range of 10m–30m indoors is common.

DC Electromagnetic DC electromagnetic fields have been used in many high-precision positioning systems. While the signal propagation speed is high range is limited to 1m–3m. These signals are very sensitive to environmental interference from a variety of sources including the earth's magnetic field, CRTs, and even metal in the area. Thus, systems based on these signals need precise calibration in a controlled environment. Such systems are prohibitively expensive in practice and will not be discussed further in this report.

Ultrasound Ultrasound signals are becoming more common in positioning systems. The relatively slow propagation speed of sound ($\sim$ 343m/s) allows for precise measurement at low clock rates, making ultrasound based-systems relatively simple and inexpensive. The signal frequency is limited by human hearing on the low end and by short range on the high end. A keen human can hear 20KHz sounds. Typical systems use a 40KHz signal. Conveniently, standard sound cards have a 48KHz sampling rate—sufficient for $\sim$ 1cm resolution distance measurements. Ranges of 3–10M have been reported. Environmental factors have substantial but not prohibitive effects on ultrasound propagation, particularly speed. Humidity can slow ultrasound by up to 0.3%. More drastically, a temperature rise from 0° C to 30° C alters the speed of sound by 3%. Finally, ultrasound reflects off most indoor surfaces. Empirical studies show that 40KHz ultrasound signals reverberate at detectable levels for at most about 20ms.

2.4.2 Other technologies

Optical Optical systems range from laser-ranging-based systems to wall- or mobile-mounted video. Safety issues preclude laser systems in typical pervasive computing environments. Omnipresent video cameras also raise many issues including unintended uses. While prices have been falling, cameras remain too expensive for wide-scale deployment. Furthermore, environmental clutter in the visible light spectrum requires extremely processing-intensive scene analysis methods for positioning, leading to further expense.

Inertial Inertial systems use orthogonal gyroscopes and/or accelerometers mounted on mobile elements to measure movement from a known initial position. Conceptually, acceleration is integrated to find velocity and integrated again to find position. Unfortunately, errors accumulate over time. Without recalibration, positioning error is unbounded. Over short time periods, inertial systems can be quite accurate.

3 System Properties

There are many ways to architect locations systems based on the techniques and technologies discussed in Section 2. The different design points represent trade-offs in the space of overall system properties. We have discuss six such properties below: symbology, errors, location rate, scale, cost, and centralization. Much of the taxonomy in this section follows an excellent survey of location systems by Hightower and Boriello [HB01a, HB01c].

3.1 Symbology

The fundamental characteristic of a location system is the symbology used to label locations. A wide variety of symbolic systems might be used. For example, an architectural floor plan can be used to generate labels for rooms, hallways, stairwells, etc. A location system can then assert that some piece of hardware resides in a particular room.

The most common symbologies are analytic geometry coordinates systems (*e.g.*, Cartesian coordinates). Analytic geometry systems have three key properties that recommend them for use by location systems. First, coordinate systems are general. Using additional levels of abstraction, any locus of points in the coordinate system can be considered an element of some other labelling system (*e.g.*, a room in a floor plan). Second, coordinate systems have arbitrary precision. Since a label is just a tuple of numbers, system designers can select any number of equivalence classes (consider any number of significant digits) for each element of the tuple. Third, coordinate systems are readily convertible. That is, given a (locus of) point(s) described in one coordinate system, it is nearly trivial to generate the description of the same point(s) any other coordinate system.

Until now, we have implicitly assumed that elements to be approximated by points. However, additional location information can be associated with the elements by modeling their extent or orientation.

3.2 Errors

The error characteristics of measurements and the resulting error characteristics of the calculated location limit the set of applications a location system can support. Errors are classified as random or systematic.

Random errors are usually modeled by a Gaussian probability distribution. The greater the variance, the lower the *precision*. The degree to which the random variation is centered on the true value is the *accuracy* of the system. Accuracy and precision are often reported together as the size of a confidence interval. For example, with probability 0.95, an estimate is within 1m of an actual location.

Systematic errors introduce *bias* into the measurement, reducing accuracy. Qualitatively, we say a measurement is not *robust* or resilient to some perturbation of the system if bias results. For example, known causes of bias in infrared systems include interference from sunlight and shadowing (obstructions). Since IR measurements are not robust to these causes, we can choose to separate out measurements in the presence or absence of these biases, giving separate error distributions. Alternatively, if it is valid to model the occurrence of these biases with a probability distribution, we can account for their occurrence with decreased precision.

In addition, Hightower *et al.* identified *dilution of precision* (DoP) as an important metric for geometric-based location systems. The method of combining low-level measurements affects the way errors are propagated to higher levels of abstraction. DoP is a unitless factor summarizing the quality of aggregate geometric measurements. If measurement errors are uncorrelated and have common variance then DoP is a function of the geometric arrangement of the sensors. For example, when combining two ranging estimates with equal uncertainty, the resulting location uncertainty is smaller when the ranges are orthogonal than when the form an acute angle. DoP is a quantitative measure of this effect.

3.3 Location Rate

The rate at which locations for system elements can be calculated affects the set of applications the system can support. The dual of location rate is location lag — the delay after a system element moves before that movement is reported by the system. (Lag can also be considered a timing bias in positioning.) Virtual reality (VR) applications require very fast update rates. People wearing VR displays report motion sickness from as little as 10ms lag. At the opposite extreme, many businesses have thrived with only weekly or even quarterly inventory reporting.

3.4 Scale

The archetypical question about any system is "Does it scale?" In other words, what is the system behavior as it gets "bigger"? Location systems need to scale on two axes: geography and density. Geographic scale measures the area or volume covered. Density measures the number of elements to be located per unit of geographic area per time period. As more area is covered or more elements are crowded in an area, more support infrastructure may be required, signalling channels may become congested, or more calculation needed to compute locations. Scaling up a system may be prohibitively expensive or require redesign.

3.5 Cost

Costs of a system can be measured in many ways. Important costs include time, space, weight, and, of course, money. The time cost of a system can include installation and maintenance. Mobile units have very tight space and weight budgets. Base station density can also be a space cost.

Monetary costs can be measured in many ways. First, there are salaries (*e.g.*, the cost of personnel to perform installation and maintenance). Second, there are incremental material cost (*e.g.*, the cost of additional mobile units or infrastructure elements). Incremental costs may be computed per additional system element or per unit of additional coverage area. Of course, cost accounting can be difficult, because some costs may be considered sunk costs. For example, a location system layered over a wireless network may be considered to have no hardware cost if all the necessary elements of that network have already been purchased for other purposes.

Energy is another cost of a system. Energy budgets on mobile units are every bit as tight as size or weight budgets. Practically, the energy budget of a mobile unit is measured by the operational lifetime of system elements between rechargings.

Some mobile units (*e.g.*, electronic article surveillance tags widely used in retail and library settings) are completely energy passive. EAS tags only respond to external fields and, thus have an unlimited lifetime. Other units (*e.g.*, laptops) have a lifetime of only several hours. Note that laptops are general purpose computing devices and will be deployed with or without a location system. In this case, the relevant question is the percentage *change* in operational lifetime. In the case of EAS tags, or other location tags attached to unpowered objects, it is the absolute operational lifetime of the tag itself that matters.

Finally, energy consumption by need not be constant. Many energy saving techniques (*e.g.*, periodically powering down a transceiver) can be part of a system design.

3.6 Centralized vs. Localized Computation

After collecting sensor data, location systems perform some amount of computation to assign location(s) to system elements. That computation may be performed by a system element that needs to know its own location, by some collaboration of system elements, or by a centralized infrastructure.

Conceptually, any sufficiently powerful processing element with recent and sufficient data can compute the location of any element in the system. However, the question of whether to localize location computation has large effects on the engineering and design trade-offs in the system.

From an engineering viewpoint, the question is: where is the processing power and how do we get the data to it? These have implications for such fundamental costs as such as size, weight, and energy budgets, circuit complexity, and channel congestion.

From a design perspective the questions raised include how are system elements named, what applications need to know location information, what rights pertain to the control of location information, and what trust model can be used to support those rights?

4 Survey

Having identified the common measurement and positioning techniques and identified the important properties of location systems, we are now in a position to discuss specific systems. We begin by reviewing the well-known Active Badges system before discussing several newer systems. We concentrate on five systems: AHLoS, RADAR, Cricket, Cricket Compass, and Active Bats. In addition we briefly discuss, commercial systems from PinPoint, Locus, and Lucent as well as the government-sponsored GPS and E911 systems. Finally, we mention a number of other systems concerned with providing indoor locations.

4.1 Active Badges

The Active Badge system by Harter *et al.* has the distinction of being the first indoor location system targeted at pervasive computing [WHFG92, HB93]. Active Badges is a proximity-based system built over a bidirectional 1KBaud infrared data link. One infrared access point was placed in each office. The mobile unit is a small, lightweight infrared transceiver that broadcasts a *globally unique identifier* (GID) every 15 seconds. Since infrared signals reflect off nearly every indoor surface, GID broadcasts are easily contained in an office, providing highly accurate localization at room granularity. Unfortunately, reflection also means the receiver can derive little or no directional information.

Location information is collected in a centralized geographic database called the location server. The location server supports five sorts of queries. Specific badges can be located currently. The badges collocated with a specific badge can be identified. The badges currently in a specific room can be identified. A long lived query can notify the user when a particular badge is located. Finally, a history of the locations of a named badge can be located.

The applications supported are a range of *follow-me* services. Examples include automatic telephone call forwarding and "teleporting" of a user's desktop to a computer in the current office. The Active Badge design of a centralized location database supporting follow-me applications remains the archetype for many of the current generation of location systems.

4.2 Ad Hoc Localization System

The target environment of the Ad Hoc Localization System (AHLoS) is a large, dynamic network of cooperating, low power, tiny sensors [SHS01]. The ad hoc nature of the network requires that there be minimal installation overhead, particularly little human intervention. Savvides *et al.* considered ranging technologies and combining techniques separately. [SHS01] is more a design study than a whole, integrated design. While various design elements are sketched, no entire system is described.

4.2.1 Technologies

The authors studied radio-frequency and ultrasound ranging technologies. The most important conclusion drawn is that they were unable to obtain a robust model of RF attenuation. Bias in received signal strength models were caused by multipath, fading, shadowing effects or the height of the receiver. They were able to use RF SS measures in a controlled, open, outdoor environment (a football field). The authors report errors of 2–4m (but do not give a confidence level) at a range of about 30m.

The authors had greater success using synchronized multi-signal (ultrasound and RF) time-of-flight ranging. AHLoS *Medusa*[2] nodes emit simultaneous RF and ultrasound signals. Since the effective range of signals is 3m, the ToF of the RF signal is discounted. They report errors of less than 2cm.

4.2.2 Techniques

AHLoS uses multilateration to assign coordinates to nodes in the horizontal plane. Nodes are classified either as *beacons* (that have known positions) or *unknowns* (that do not). The authors use three variants of multilateration: *atomic*, *collaborative*, and *iterative.*

Atomic multilateration Savvides *et al.* distinguish the simple case considered in Section 2.3 of a single unknown in range of three or more beacons, calling it atomic multilateration. As noted above, additional beacons can be used to solve for the speed of sound, and reduce error.

Collaborative multilateration A network of beacons and unknowns with ranging information can be completely constrained even if no unknown has three beacon neighbors. In such a network, the known quantities (beacon locations and ranging information) are collected together and the system is solved for the unknown locations using optimization methods such as gradient descent and simulated annealing. Collaborative multilateration is the process of identifying and solving a completely constrained network.

Unfortunately, the authors fail to correctly identify the cases in which such a completely constrained system exists. The definitions given by the authors have two distinct problems. First, they define a condition on individual nodes that is actually a property of graphs of nodes. Second, even a reasonable graph-oriented interpretation of the definition (based on pseudocode in the paper), yields a property that is not complete. That is, there are completely constrained graphs that do not fit their definition. We model networks of beacons and nodes as graphs where each node is annotated by a pair of Cartesian coordinates (possibly unknown) and each edge has a weight equal to the Euclidean distance between nodes.

DEFINITION 1 *A node is a* participating *node if it is either a beacon or if it is an unknown with at least three participating neighbors.*

We replace this ambiguous node-oriented definition with following two definitions:

DEFINITION 2 *A boolean function p of the nodes of a graph G=(N,E) is a* participation function *iff*

$$\forall n \in N \begin{cases} p(n) \Leftarrow & n \textit{ is a beacon} \\ p(n) \Leftarrow & [\exists x, y, z \in N, \text{distinct s.t.} \\ & p(x) \wedge p(y) \wedge p(z) \wedge \\ & \{(x,n),(y,n),(z,x)\} \subset E] \end{cases}$$

DEFINITION 3 *A graph G=(N,E) is* participatory *iff there exists a participation function f s.t. $\forall n \in N, f(n)$ is true.*

Unfortunately, even this definition is insufficient. Consider the following counterexample. Let G be K_4 where all nodes are unknown (but the ranges between all pairs of nodes are known). Note that this graph is participatory under the constant participation function *true.* (In fact, every graph is participatory under the constant *true* function.) However, without *any* beacons, it is clearly not possible to align these unknowns with any predefined coordinate system. Further, notice that the constant function *false* is also a participation function on K_4. In this case we would rather discover the more restrictive function.

---

[2] The authors incorrectly identify Medusa as a multi-headed monster from Greek mythology. (That description fits the Hydra.) In fact (or myth), Medusa, Sthenno, and Euryale were three sisters known as Gorgons who were said to have asps growing out of their heads instead of hair. Even so, the name fits the appearance of the node. An array of ultrasonic receivers stand on wire stalks weaving their way to face in an open direction.

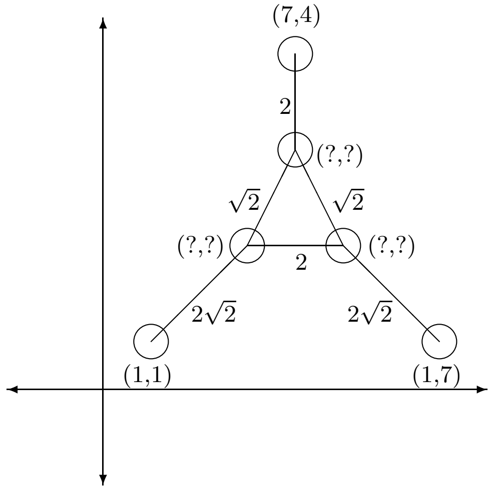

Figure 1: A graph that is not collaborative but completely constrained in 2-dimensions.

What the author's algorithm actually finds is the *minimal* participation function on a graph. That leads us to the following claims and definitions:

CLAIM 1 *For any graph $G = (N, E)$ there is a minimal participation function $\bar{p}$, where the minimality is determined by containment on the sets of nodes selected by the function.*

DEFINITION 4 *A graph G=(N,E) is* collaborative *iff there exists a minimal participation function f s.t. $\forall n \in N, f(n)$ is true.*

CLAIM 2 *The positions of nodes in any nonempty collaborative graph are completely constrained (in two dimensions).*

We are now able to show that the method of finding and solving collaborative graphs is not complete. Consider the graph in Figure 4.2.2. The minimal participation function (and the one the author's algorithm actually finds) includes only the beacons. However, there are six unknowns (three x-coordinates, three y-coordinates) and six constraining equations (from the six ranges).

Iterative multilateration Once an unknown node has been located, it acts as a beacon. Therefore, there can be value in iterating multilateration in rounds, using atomic multilateration and collaborative multilateration as primitives. In each round, a completely constrained subgraph containing at least one unknown is selected and all unknowns in it are located. After each round, the newly located nodes become beacons. Thus, location information propagates across the network.

4.2.3 Simulation studies

To evaluate their algorithms, Savvides *et al.* performed a series of simulation studies. They model ad hoc networks as networks of beacons and unknowns placed in an area uniformly at random. While this model seems prevalent in the ad hoc networking literature, its validity is questionable. In particular, in the indoor environment, it seems unlikely that beacons will be distributed randomly. A more likely scenario is that a few initial beacons will be placed strategically.

One problem with iterative multilateration is that errors propagate along with location information across the network. Measurement error (in ranging), initialization error (in beacon placement), and dilution of precision error all propagate [HB01b]. In a simulation study with 5 beacons and 45 unknowns placed randomly in 15m x 15m square, the authors found that errors can grow quite rapidly. The initialization and ranging errors were both modeled as 2cm Gaussian distribution. In both cases, estimated position error was as large as 20cm.

A second study showed that collaborative multilateration can substantially increase the percentage of unknowns that can be located in a sparse-beacon random network. If the network is relatively sparse (200 nodes, 20 beacons, in a 100m x 100m square) the percentage of unknowns located can be nearly tripled.

The authors also studied the issue of localized vs. centralized computation. In their model, all communication travels over wireless RF links. As a result, centralized localization computation incurs large penalties in channel and energy utilization. The results show that when atomic multilateration is possible, there is nearly an order of magnitude savings in network traffic and energy expenditure per node when computation is performed locally.

4.3 RADAR

In design of the RADAR system, Bahl *et al.* seek to enhance the value of commodity IEEE 802.11 wireless networks [BP00b, BP00a]. The location system uses no custom hardware. In these networks, an RF transceiver (base station) acts as a bridge between the wireless and the wired networks. Mobile units (usually laptop or handheld computers with 802.11 cards) communicate directly with the base station. Base stations are assumed to be stationary and continuously connected to both data and power networks. Periodically, base stations broadcast beacon signals (including the identity of the source base station) that are measured by the mobile units. [3] In the experiments described, base stations are assigned separate channels and synchronized. Beacon broadcasts are scheduled five or ten times per second and scheduled so as not to collide. In addition, broadcasts are schedule such that mobile units have time to switch between channels to pick up all (3 or 4) beacons. All base stations could be received throughout the entire test area.

Location is performed by scene analysis of the RF signal strength characteristics.[4] The basic RADAR location method is performed in two phases. First, in an off-line phase, the system is calibrated and a model is constructed of received signal strengths at a finite number of locations distributed about the target area. Second, during on-line operation in the target area, mobile units report the signal strengths received from each base station and the system determines the best match between the on-line observations and any point in the off-line model. The location of the best-matching point is reported as the location estimate.

4.3.1 Off-line model construction

RADAR uses a fairly simple model of signal characteristics. A set of reference points is selected. (In the experiments described, 70 reference points were picked in 980m^2 area of one floor of an office building.) For each point, a tuple is stored containing the point's coordinates and a signal strength value for each base station in the system. The tuple can be interpreted as a prediction of the signal strength a mobile unit situated at that point will receive from each base station. This collection of tuples is call the *signal space*.

Empirical Model To create an empirical model, the authors placed a laptop at each of the 70 reference points and measured the received signal strength over several seconds. The recorded values were averaged and stored.[5]

Since variations in environmental conditions including the movement of large numbers of people have large effects on received signal strength, multiple models can be developed by taking measurements under each type of variation. In addition to the usual tuples, the base stations record signal strength statistics for each other. Thus each base station knows the mean μ_e and standard deviation σ_e of the signal strength received from every other base station in each environmental condition e.

Analytic Model The authors adapt the Floor Attenuation Factor propagation model (FAF) as defined by Seidel to predict the received signal strength at reference points and thus to create signal space [SR02]. The model accounts for signal loss due to distance and attenuation of signals through obstructions. The adapted model depends on the *Wall Attenuation Factor* (WAF).[6] The value for the WAF was determined by regression analysis from empirical observations of signal strength through various numbers of walls at various distances. Other factors in the model determined empirically include the power of the transmitter (averaged across base stations) and the exponent of loss due to distance.

To generate the signal space database, the authors use floor plan information to determine the distance from each base station to each reference point and the number of intervening walls. These parameters are then plugged into the model and a signal strength prediction is made for each base station/reference-point pair.

4.3.2 On-line location matching

The authors develop three algorithms for determining locations from observed positions. The basic algorithm is called *nearest neighbor in signal space* (NNSS). When a mobile unit location is to be calculated, the beacon signal strengths are recorded. The algorithm then finds the point in the model signal

---

[3] In fact, the authors describe two different configurations of the system in the two papers. One where the base stations broadcast beacons and one where the mobile units broadcast. They found little asymmetry.

[4] The authors claim the system uses triangulation but, as we shall see, no explicit ranging or angulation is performed.

[5] 90° orientation information is also recorded, but is not reported in the model.

[6] In this case the obstructions are walls rather than floors, thus the name change.

strength database that minimizes the Euclidean distance between the observed signal strength values and the predicted values. The location of the nearest neighbor is reported as the current location of the mobile unit. NNSS-AVG is a simple variant of NNSS that reports the average position of k nearest neighbors in signal space as the location of the mobile unit.

A third, more complex, *Viterbi-like* algorithm uses the history of location estimates to reduce *aliasing.* Aliasing occurs when two physically distant locations demonstrate common signal strength behavior and, thus are close in signal space. During each positioning calculation, the k nearest neighbors in signal space are determined ($k-$NNSS). In addition, a history of the last h such $k-$NSSS sets is maintained. These hk points in tuple space are treated as a graph. If $t_{i,j}$ is a tuple in signal space for $1 \leq i \leq h$ and $1 \leq j \leq k$, then the edges of the graph are $(t_{i,j}, t_{(i+1),j'})$ for $1 \leq j' \leq k$. That is, there is an edge between every pair of tuples in successive $k-$NNSS sets. Each edge is assigned a weight equal to the Euclidean distance between the *physical* locations associated with its vertices. The idea is to model the likelihood of a user moving between the two locations in successive time periods. Finally, to compute a location estimate, the shortest path in the induced history graph is computed. The endpoint of the shortest path in the oldest $k-$NNSS set is selected as the tuple specifying the mobile unit location.

To select among multiple signal space models, the authors exploit the fact that base stations have fixed locations. At runtime, the mean received signal strength over a sliding window is computed. Assuming a Gaussian error distribution, the probability that the measurements correspond to each environment is computed. The model of the environment with the highest probability is selected.

4.3.3 Simulation results

The authors report the results of simulations of their algorithms. Unfortunately, the methodology is fairly suspect. A reference point is selected from the calibration data and the signal space is constructed from the other 69 points. The algorithm is then run. However, the selection of the 70 points was far from random initially. They were spread along the corridors of the office space. (No in-room positions were used.) While the points were not evenly spaced, it is not at all clear that the positional relationship among these 70 points reflects the relationship of random or typical user locations to a set of fixed reference points. Furthermore, the method guarantees that the modeling environment is identical to the simulation environment. With these restrictions, the authors report NNSS algorithm using the empirical model generates 3.6m median error.[7] The analytic model resulted in median errors of 4.3m.[8] Using NNSS-AVG resulted in modest improvement for $k = 3$ with median error of 3.3m. The Viterbi algorithm for $k = 3$ and $h = 6$ gave substantial improvement. The median error was 2.4m while the 90th percentile error fell from 7m to less than 4.5m. Of course, this improvement comes at the cost of a six-update location lag.

4.4 Cricket

Whereas Bahl *et al.* designed RADAR to enhance the value of a particular networking technology, Priyantha *et al.* explicitly designed the Cricket system to be independent of the primary data networking technology in use [PCB00]. The designers identify four other key design points that mesh with decoupling from networking technology: decentralization, privacy, room-sized granularity, and lost cost. The system is decentralized in that each component of the system — whether fixed or mobile — is configured independently. No central entity is used to register or synchronize elements. User privacy is maintained by allowing mobile elements to compute their location locally without any outside communication. Mobile elements may then use that information locally or choose to advertise it to higher-level, remote services. For the target services, Cricket need only name the room in which a mobile element resides. Cricket achieves over 95% accuracy at this relatively low level of precision. With these specifications, Priyantha *et al.* report achieving a $10/unit (fixed or mobile) price point. In addition, since only one stationary beacon unit is required per room, the cost of scaling the system geographically is quite reasonable.

Cricket computes distances using the TDoA of synchronized RF and ultrasound signals. Each beacon emits an RF pulse uniquely identifying the space it occupies. Mobile units compute the distance traveled

---

[7] Different results are claimed for NNSS in various places in the two papers. It is not entirely clear what the variables are, but they include two separate installations in different buildings.

[8] The test set for the analytic model is unspecified. I assume it is also the data collected to form the empirical model. If so, that cross validation is a much tougher test of the model. That difference alone might explain the poorer performance of the analytic model.

by each beacon signal it hears. For each beacon, the mobile unit sorts the distances into ten inch[9] increments and counts the number of signals it hears in each increment. The distance increment corresponding to the mode of the distribution over a sliding window of samples is selected as the distance to a beacon. Cricket then reports the location of the mobile unit as the space advertised by the beacon with the smallest reported distance.

Since beacons are configured independently and do not directly communicate, there is the potential for beacon signals to interfere with each other. Cricket avoids extensive interference by introducing randomization. Beacons choose the delay from one signal broadcast to the next uniformly at random from within the interval $[150, 350]$ms. Thus, while each beacon has an average frequency of four broadcasts per second, the broadcast times are statistically independent.

A second difficulty of the Cricket design derives from the fact that the RF signals used have greater range than ultrasound signals. Furthermore, the RF signals are able to pass through obstacles such as walls, while ultrasound is reflected. The result is that it is possible, when two beacons in proximate rooms broadcast nearly simultaneously, that a mobile receiver will associate the RF signal of a beacon in the adjacent room with the ultrasound signal from the beacon collocated with the receiver. As a result, the perceived TDoA can be very small and the mobile unit can erroneously assign a very short range to the beacon in the adjacent room.

To avoid this scenario, the Cricket design requires the RF signal duration to be long enough so that the corresponding ultrasound signal arrives *while* the RF signal is still being received. That is, if the RF signal has a transmission rate of b bits/s and the maximum propagation time for an ultrasonic signal is τ seconds, then the length of the RF signal is selected to be at least $b\tau$ bits long. Furthermore beacons are carefully placed equidistant from location (room) boundaries to alleviate border ambiguities.

In experiments with the system, Cricket units were able to correctly identify the room in which they were located in over 95% of cases when stationary, even when only one foot from room boundaries or three feet from interfering RF sources. In the tests of units moving between spaces at walking speed, similar results were obtained when the sliding window size of the modal algorithm was set at five samples. Rooms are required to be at least four foot square.

4.5 Cricket Compass

Cricket Compass is an extension of Cricket that infers orientation to within a few degrees and position on a coordinate grid to within an average error of 6cm [PMBT01]. The extra precision needed to determine orientation drives the change from Cricket's proximity-based system to a coordinate-based system. Cricket Compass continues to use (a more complex) TDoA of synchronized ultrasound and RF signals to perform ranging. To determine coordinates, Compass uses multilateration from at least four beacons in line of sight of the mobile unit. Beacons must now be configured with precise coordinates rather than merely room identifiers. The greater density of beacons (at least four per room) increases the likelihood of beacon interference, although the authors do not analyze this probability.

Cricket Compass preserves many of the design goals of Cricket. Location calculation can still be performed at the mobile unit, preserving user privacy. Compass continues to be decoupled from general data networks. The cost of the system is somewhat greater than that of Cricket. Each mobile unit now contains an array of five ultrasonic receivers spaced over several centimeters. In addition, as mentioned above, several beacons have to be precisely calibrated in each room.

Cricket Compass uses an extremely clever arrangement of ultrasonic receivers to determine precisely the differential distance of the receivers from the beacon source. In fact, the distances measured are smaller than the receivers and even smaller than the wavelength of the signal. The five receivers are placed in a "V" shape. The two legs of the V are orthogonal and each consists of three receivers. (The receiver at the point of the V is common to both legs.) The distances between the receivers on each leg are carefully measured to be relatively prime multiples of $\lambda/2$ where λ is the wavelength of the ultrasound signal. The upshot is that the receivers can measure the *phase difference* for each pair of collinear receivers. This information together with the height-to-distance ratio $(z/\bar{d})$ of the unit to the beacon generating the signal is sufficient to compute the orientation of the V with respect to the V to beacon line.

The height and distance to the beacon is computed

---

[9] The use of English units by the authors is baffling.

using multilateration from the ranges to four beacons. Cricket Compass computes the (x, y, z) coordinates and v^2, where v is the speed of sound. The domain restriction that beacons are on the ceiling and mobile units are below them resolves the remaining ambiguity of which root to take. A unique set of values satisfy the constraints if the four beacons are placed neither on the same line nor the same circle.

The authors analyze the error potential of their orientation algorithm. By taking the differential distances and the height-to-distance ratio as two independent measurements (and therefore sources of random error), the authors are able to express fractional error ($\frac{\delta\theta}{\theta}$) as a function of the differential distance and the $z/\bar{d}$. Notice that the differential distance is proportional to the angle θ (the more oblique the angle, the closer one receiver and the farther away the other.) Similarly, $z/\bar{d}$ becomes large as the mobile unit moves directly under the beacon. The authors note that since four beacons must be visible for multilateration, it is usually possible to pick a beacon for orientation measurement with small θ and $z/\bar{d}$.

Experiments validated this model. Average orientation errors grew from less than 3° for 10° angles up to 15° for 80° angles. However, errors remained under 5° for angles of less than 40°.

Mean location error was 6cm when the mobile unit was placed near the center of the room. However the system was found to be less robust to ultrasound reflections close to walls, yielding errors of up to 25cm. The authors suggest the addition of a fifth beacon to detect such reflection errors.

4.6 Active Bats

The Active Bat location system is a follow-on to the Active Badge System [WJH97, War98, HHS+99]. It builds on the Active Badges model of a centralized location tracking system informed with data from beaconing mobile units. The mobile Bat units are quite small.

Like Cricket, Bats use TDoA of synchronized RF and ultrasound signals to perform ranging. Like Cricket Compass, Active Bats use multilateration to compute the coordinates of the mobile element in relation to the known locations of ceiling-mounted units. However, the centralized nature of the Bat system led Ward *et al.* to a significantly different set of design choices from those of Priyantha *et al.*

First, the ceiling-mounted units are ultrasound *receivers* for the signals emitted by the Bats. Second, receivers are connected by a wired, powered network and organized hierarchically system wide. Second, to alleviate contention for the ultrasound channel, the system is polling based. Each Bat registers with the system (using an RF side channel) upon arrival in a covered area. Periodically, a base station broadcasts an identifier as an RF signal. Simultaneously, the base station sends a synchronization pulse via the wired network to all the connected receivers. When a Bat hears its GID broadcast, it responds with an ultrasonic signal. The TDoA between the wired synchronization pulse and the Bat ultrasound broadcast at the receivers is used for ranging and multilateration. Third, multipath effects are avoided by spacing out the polling slots by at least 20ms to allow any ultrasound reverberations to die out. Fourth, to deal with shadowing, the system first filters out measurements that violate the triangle inequality and then throws out statistical outliers. Fifth, explicit cell handoff is required. When a Bat moves out of the coverage area of one base station into that of another, it must reregister. Finally, rather than using additional base station measurements to calculate the speed of sound, the system uses temperature sensors and an analytic model to adjust the ranging parameters. All base stations must be synchronized and use time division multiplexing of the RF channel to deal with overlap.

Experiments were carried out in a test installation in two rooms covering 280m^3. Using an array formation that spaced receivers at 1.2m intervals, 100 receivers were wired into the ceiling. In 100,000 measurements, 95% of readings had errors of less 9cm. In simulation, the statistical elimination of reflected signals had a 10% false negative rate (failed to eliminate a reflected signal) and a 4% false positive rate (erroneously eliminated a valid measurement). Bats have an operational lifetime of several months even when located several times per second.

Orientation measurement was also studied. Two methods were used. First, two or more Bats were attached to a rigid object and the difference of their calculated position used to estimate orientation. Unfortunately, since the basic location system has a 95% confidence interval of 9cm, the random error can overwhelm the actual spatial difference between two Bats. Experiments show up to a 15° error in 90% of measurements with a 22cm separation between the Bats. When the separation is reduced to 6.5cm, the 90th percentile error rises to as much as 50°.

The second orientation method uses the fact that most objects to which a Bat is attached are opaque to ultrasound. Thus, the set of ultrasound receivers in the shadow of the object depends on orientation of the object. The authors devised a method to determine the orientation from the shape of this shadow. The performance of this method was reported as somewhat worse than the result from the 6.5cm separation experiment.

4.7 PinPoint

3D-iD is a commercial location system positioned to compete with retail EAS systems [WL98]. Pinpoint uses RF round-trip times to do ranging. Like Active Bats, it uses an installed array of antennas at known positions to perform multilateration. When a mobile tag receives a broadcast, the tag immediately rebroadcasts it on a different frequency, modulated with the tag's identifier. A cell controller cycles through the antennas, collecting a set of ranges to the tag. Using a 40MHz signal, the prototype system achieves 30m range, 1m precision, and 1 year lifetime. Tags are about the size of a "double-thick credit card" and have an update rate of once per five seconds. The rate is kept low to extend operational lifetime.

Evidently, the base station broadcasts a signal nearly continuously through some antenna. Tag interference is reduced by the fact that mobile units sleep for 4.995s and then wake up and (re)broadcast for 5ms. However, as the sleep cycle is deterministic, periodic beat effects can result. The authors only say that the controller can anticipate collisions and the "absence of expected collisions indicate that a tag has moved out of range."

4.8 Wide Area Systems

While the scope of this paper is indoor location systems, two wide area systems are so well known that they should be mentioned: GPS and E911. We also mention two commercial systems surveyed by Koshima, Locus and server-assisted GPS, that attempt to bridge the gap between wide area, outdoor systems and indoor location systems [KH00].

GPS The global positioning system is probably the most widely known automated positioning system. A system of transmitting satellites provides worldwide coverage. Anywhere a mobile receiver can obtain line of site to four of the satellites, it can locally perform a multilateration computation with average estimated error of 35m in typical civilian use. GPS is unique in using multiple, synchronized sources with known locations (the satellites) and a single receiver with unknown location. The satellites use atomic clocks to maintain synchronization and precise models of satellite motion to predict satellite positions at the time of broadcast. Unfortunately, GPS signals do not penetrate well into urban environments. The transmitted signals are weak enough that they are blocked by most walls and even stands of trees.

E911 The FCC has mandated that all mobile telephone vendors be able to locate the mobile units in case of emergency. The FCC will require receiver-based techniques to locate 95% of calls with 150m and transmitter-based techniques to perform the same task to a precision of 300m. Many approaches to the problem are being taken by vendors, including antenna proximity, angulation, multilateration via signal strength and time of flight, as well as GPS-enabled handsets.

Server-Assisted GPS Among others, Lucent has announced a system that uses a stationary server to assist indoor mobile receivers to acquire GPS signals. The base station continuously tracks the GPS satellites via some well-placed antennas. When a mobile unit needs to be located, it obtains information from the fixed server that, in effect, enhances the mobile unit's sensitivity to the relevant GPS signals, enabling it to collect enough information within 1s for its position to be calculated. Furthermore, the system is inherently a differential GPS system. Differential GPS systems are able to eliminate some of the errors purposefully introduced into the civilian system. Lucent has reported outdoor location errors of less than 5m.

Locus The Locus system uses RF signal-strength sensing and scene analysis to locate specific PHS wireless devices [KH00]. Locus is overlayed on the basic PHS cellular service. To refine location beyond cell proximity, Locus uses a signal propagation model to account for some multipath effects. They report a mean error of 40-50m.

4.9 Other

Several other groups are designing and investigating location systems based on similar technologies. Girod *et al.* use combined RF-ultrasound TDoA methods similar to Cricket and Active Bats [Gir00, GE01]. Bulusu *et al.* have investigated RF signal-strength-based location methods in ad hoc settings [BHE00, BHE01]. The SpotOn project also proposed an RF signal-strength model of ranging [HVBW01]. The LocustSwarm is an infrared proximity system using technology similar to that of Active Badges and a system architecture like that of Cricket [KS97]. An interesting feature of the system is that users can place annotations in spaces in addition to the basic beacon identifier.

Different approaches were taken by the Smart Floor and PFinder projects. The Smart Floor project uses the characteristics of walking patterns to identify and locate users in contact with sub-floor sensors [OA00]. The PFinder video system tracks the body movements of a single individual in a video image. The processing involved is intensive enough to require special purpose hardware to support real-time recognition [WADP97].

5 Discussion

We now evaluate the systems we introduced in Section 4 in terms of the properties we identified in Section 3. We pay particular attention to the trade-offs among AHLoS, Cricket, Cricket Compass, and Active Bats.

5.1 Symbology

The choice of symbology is essentially one of level of abstraction and precision. While applications may require a wide variety of location representations, it is unlikely these systems will be directly convertible. Furthermore, it is clear that as technology and techniques improve, there is a trend toward increasingly precise coordinate systems. For example, Active Badges and Cricket, examples of successful proximity systems using architectural-oriented symbologies, have been followed up with the coordinate-based systems Active Bats and Cricket Compass.

The most attractive solution is a layered approach in which high-level symbologies are built over coordinate representations. The coordinate system should be capable of matching the precision available from the hardware and algorithms used. The structure of high-level symbologies should be driven by application-level semantics. Of course, if the location system is capable of detecting orientation, the coordinate system should be able to represent it.

The Active Bat system takes this approach. The Bat location database is designed to support the conversion of point locations specified in coordinates into relevant spaces (*e.g.*, a room or the space in front of a monitor). Applications can perform geometric containment queries on the database to find information of the form “a Bat has entered the space in front of the monitor.”

5.2 Error Characteristics

Unfortunately, typical standards of empirical measurement in this field are fairly low. The digital nature of the technology means that there are relatively few subfields where random error plays a large part in system performance. As a result, we have become sloppy in reporting experiment design and analysis. Playing into the weakness, location systems are inherently analog and proper evaluation requires empirical measurement. Unfortunately, due to the lack of standards, it is difficult to compare the error characteristics of these systems in the available reports. Some authors report mean error but not confidence intervals; others report several different results for the same technology without cross-experiment analysis. One key difficulty is that there exists no clear indication of what levels of precision and accuracy are acceptable for location-aware applications to function. Thus, location system designers choose their own characterization of “good enough”.

It is possible, however, to make some generalizations. First, there seem to be some basic equivalence classes. Cricket Compass, Active Bats, and AHLoS all achieve precision of 1–10cm with high probability. At the next level, Cricket, RADAR, Active Badges, and Pinpoint generally report errors in the range of 1–10m. Finally, outdoor systems such as GPS, E911, and Locus have errors of tens of meters.

Second, indoor radio frequency methods are far from robust. Signal strength measurements continue to defy the attempts of designers to develop practical models. Multipath effects and signal attenuation remain extremely difficult to predict in cluttered, variable indoor environments. Even a fairly good model of a particular space (developed at high cost) can be invalidated by the movement of people in the space. Similarly, cost-effective time-based measurement of RF signals continues to be difficult. Even using round-trip times to avoid clock synchronization does not give good error performance.

Third, while time-based ultrasound measurements give a one to two order of magnitude improvement over RF methods, the systems continue to be susceptible to shadowing effects. So far, most systems have finessed this issue with ceiling-mounted units. However, a more generally robust method should be found.

Finally, more work is needed to identify the correct way to report orientation error. Ward *et al.* modeled all Active Bat orientation error as equivalent and random. Priyantha *et al.* developed an analysis showing a fundamental proportionality of angle size and error. While the error levels they reported were quite reasonable, it would be nice to see a better way of reporting error as a function of angle size.

5.3 Update Rate

Potentially, the faster propagation speed of RF systems could be translated into faster update rates. However, at the moment, the constraining factors reported are processing speed (including operating-system-level issues) and mobile unit energy budgets. Ultrasound systems encounter problems at rates greater than 50 updates per second, due to reverberations. To pass the 50Hz rate, ultrasound systems will have to employ more complex signal processing techniques.

As with error levels, application requirements for update rates remain unclear. Currently, VR-quality systems (< 10ms lag) seem out of reach. The systems surveyed pick design points for updates ranging from five updates per second to one update per five seconds.

However, citing the raw update rates is somewhat deceiving as different numbers of units can be located in each update. In Cricket, every mobile unit can update its location each time it hears a beacon. Active Bats is a polling system where one mobile unit per RF cell is located per round. Cricket Compass mobile units need to hear four beacons (*i.e.* four rounds of broadcasts) to compute a complete update, although orientation can be computed from just one. RADAR can be configured either with the base stations beaconing (like Cricket) or with the mobile units beaconing. As Bahl *et al.* note, the latter decreases update rate and raises channel contention. The AHLoS authors did not address update rate due to the preliminary nature of their design.

The update rate is most important for orientation-reporting systems. In particular, since Active Bats computes orientation at a level of abstraction above coordinate location estimation (for the multi-Bat method) there is at least a 20ms lag between the measurements of the two ends of the baseline for orientation calculation, resulting in greater error for orientation estimation during movement. Cricket Compass computes multiple locations from a single broadcast, substantially decreasing the effects of movement on the calculation.

5.4 Costs

As (almost) all the systems studied are research prototypes, projecting the costs of a method can be hazardous. However, some features of the costs do stand out. Furthermore, prototypes generally do provide upper bounds on costs, so proofs of concept for low-cost techniques are valuable.

Time Perhaps the most reliable projections of cost can be made about the installation and administrative overhead of the systems.

Active Bats centralized model and arrays of wired sensors create extremely large startup costs. The arrays not only have to be wired in place but the antennae positions carefully measured and RF characteristics calibrated. Furthermore, the centralized location database is a large and complex piece of software. Each Bat has a proxy software object maintained in the system, so every new unit must be registered with the system. The system tracks a great deal of information including location histories, temperatures, and other non-location information (*e.g.*, mouse and keyboard activation). Finally, since all location broadcasts are centrally scheduled, there are ongoing maintenance costs for the cell controller and antenna infrastructures.

Cricket Compass also requires the installation, measurement, and calibration of ceiling-mounted beacons. However, beacons are battery powered and autonomous rather than wired. No scheduling is required due the randomization. Each need only be initialized with its position in the coordinate system and then left alone until the batteries need changing. Cricket clearly has even smaller overhead costs. Installing a Cricket beacon is only slightly more difficult than installing a smoke detector.

RADAR requires relatively little hardware installation overhead, simply install three off-the-shelf base stations. Calibration is more difficult. If an empirical model is used, measurements need to be made at every reference point. If multiple models are to be used to account for environment changes, measurements need to be made at every reference point in each environmental situation. If an analytic model is to be used, then architectural drawings must be obtained and analyzed, attenuation factors measured, and the model applied to every reference point.

The ad hoc nature of AHLoS means it has relatively low overhead. Nonetheless, Savvides *et al.* consistently assume that 10–20% of nodes are beacons initially. Some means must be used to locate these nodes initially. Human intervention seems the most likely method in the indoor environment. Furthermore, the authors assume these nodes are distributed randomly throughout the space. A more likely automated scenario would be to carefully place a number of GPS-enabled nodes (say at windows) to bootstrap the system. However, that would violate the typical ad hoc network assumption of functional homogeneity of the nodes.

Money Cricket is the only system that provide hard dollar numbers. The $10 per unit price gives a simple upper bound on hardware costs. RADAR builds on a somewhat mature technology making final system costs easier to evaluate. Despite the author's goals, the system is not really free. Typical 802.11 wireless networks are spread as thinly as possible because base stations are relatively expensive (~ $140). Requiring multiple base station coverage in the target area significantly raises the cost of the system. (In fact, the authors even suggest the use of lower cost, non-data network mini-base stations in [BP00a].) The monetary costs of the other systems either were not reported or were too difficult to forecast at the preliminary stages of prototype design.

Mobile unit size and weight After marginal monetary cost, miniaturization is potentially the largest change from prototype to production. For example, Cricket and Active Bats use essentially the same technology. However, the Bat units are about one fifth the size of the Cricket. Cricket Compass is limited by the fact that each mobile unit must have a span of orthogonal ultrasonic receivers. Currently, the Compass must be about 6cm on a side. The AHLoS Medusa design is preliminary but already fairly small except for the writhing antennae. RADAR, of course, uses standard networking cards.

Infrastructure space As with infrastructure overhead, Active Bats have the highest cost with respect to space for infrastructure. Since the antenna arrays are wired, significant access to the ceiling is needed. While at least four Cricket Compass beacons are needed in each room, placing the units on the ceiling is not difficult. Finding space for Cricket and RADAR base stations is even easier. In theory AHLoS has no installed infrastructure, but once again beacon initialization is a problem. On the other hand, once the AHLoS system is up and running, moving a beacon out of the way is costless.

5.5 Localized vs. Centralized Computation

Cricket and Active Bats represent extremes of the centralization access. In Active Bats, centralized functions include clocking pulses, scheduling Bat polling locations, multilateration computation, and storage of location data. In order for a Bat to know its location, it must query the central database. Access control is also a centralized database function. (In fact, access control at the database may be too late to safeguard information. Since mobile units broadcast pulses, a second array of sensors could pirate the signals to compute location information about any mobile host that is polled.)

On the other hand, centralization in the Cricket and Cricket Compass systems consists merely of maintaining a global name space of beacon identifiers at the time of manufacture. (Similar, for example, to MAC addresses on Ethernet interfaces.) Mobile units need not be identified at all and never emit any signal detectable to the location system. As a result, the Cricket mobile units can move about anonymously but must perform time measurement and multilateration computation locally. Additionally, ultrasound signal characteristics put a heavier energy burden on the receiver than the transmitter, further raising the mobile unit energy requirements.

In the prototype described, RADAR is configured as a centralized system. However, there is nothing in the system design that requires it. As the target platform seems to be relatively powerful laptop or handheld computers, a fair amount of storage and computation power is available. The propagation models (signal space tuples) developed in the off-line phase could be distributed to mobile units before beginning on-line operation. With that information, the mobile units could act autonomously, just like Cricket units.

AHLoS target environment is inherently decentralized. However, in the prototype testing described, location simulations were performed in a centralized manner. While it is clear that atomic multilateration can be computed locally by each unknown node, collaborative multilateration raises more challenges. The authors claim the algorithm can be fully distributed but that is not entirely clear. The algorithm appears to require identifying a consistent cut of connectivity, ranging, and beacon location information in a highly dynamic environment. One well studied method for finding a consistent cut is to invoke a group membership protocol. Unfortunately, group membership protocols are known to require a period of network quiescence or perfect failure detectors to operate. Furthermore, such protocols are relatively expensive in messaging costs. Even assuming a stable group membership, the authors suggest using fairly intensive numerical algorithms to solve for the locations of the entire group. It seems this work might be distributed in a more energy-efficient way.

5.6 Scale

The ability of location systems to scale geographically is dominated by installation costs. As previously mentioned, Active Bats and Cricket Compass have relatively high costs compared to Cricket and AHLoS. Similarly, RADAR has high costs in model building.

Centralized systems have more problems when scaling the density of their systems. That is, the more mobile units a centralized system must track, the more tight the competition for resources. In Active Bats, the channel access becomes a constraining factor. Since only one mobile unit may be located per time slot in a room (ultrasound containment area), increasing the number of units per room means decreasing the update rate for each mobile unit. A centralized configuration for RADAR has the same difficulty. Furthermore, as the number of units in the whole system grows, the burden on the centralized location database grows. In the Active Bats system, the large number of factors tracked per unit (in addition to location) only multiplies the difficulty.

On the other hand, any number of Cricket units can locate themselves simultaneously. While there may be contention for access to location-aware services, the location service itself scales.

Without some smart pruning of ranging broadcasts, AHLoS has a potential scaling problem. As the number of nodes grows, the system wide ranging calculations has the potential to grow quadratically (proportional to the number of node pairs).

6 Research Directions

While the systems surveyed have achieved a large margin of success in generating location information suitable for use in context sensitive applications, several avenues for future work seem apparent. Below we briefly examine five remaining challenges: sensor fusion, software support, distributed multilateration algorithms, maximizing beaconing rates, and bootstrapping.

6.1 Sensor Fusion

Each of the systems surveyed relies one "best" method of calculating location. As we have seen, each ranging method and each system design has shortcomings. As others have noted, combining readings from different types of sensors with different capabilities and error profiles opens the possibility of exploiting redundancies and contradictions to reduce overall location uncertainty [HBB02].

For example, the FAA is implementing an aviation navigation system that combines GPS-based navigation with inertial systems [Ome02]. Inertial systems are highly accurate over short time periods but errors accumulate without bound. GPS is highly accurate (for aviation purposes) when a mobile unit is able to lock-on to the signals from at least four satellites. The proposed system bridges blackout periods when satellite signals are unavailable (usually several seconds to a minute) using inertial measurements.

A similar arrangement for pervasive computing might help alleviate channel congestion and location update rate problems. While multilateration can occur periodically, intra-period localization can be computed using inertial measurements. Similarly, inertial measurement can be used to lower energy costs and extend operational lifetime. At the very least, it should be simple to measure the *absence* of acceleration. Thus, when an object is known to be stationary, there is no reason to expend energy on signalling and multilateration computation.

6.2 Software Support

While this paper is focused on low-level localization systems, there is clearly a need for identifying the correct software interface to such systems. A number of studies focus on software for location- and context-aware computing. Grimm *et al.* describe a comprehensive framework for pervasive computing applications including support for service discovery [GDL$^+$01]. As already discussed, Addlesee *et al.* developed a framework for building context-aware applications around a centralized location database for the Active Bats project [ACH$^+$01]. Miu describes the software architecture of a navigation system built using Cricket [Miu02]. Raman *et al.* have examined scaling security mechanisms for pervasive computing [RCB$^+$02].

The most compelling case for a location-aware computing architecture is presented by Hightower *et al.* [HBB02]. They identify five principles of location-aware computation:

1. "There are fundamental measurement types." As we have seen above, ranging, angulation, proximity, and assertion of position are fundamental to all the systems surveyed.

2. "There are standard ways to combine measurements." We enumerated the most common ones in Section 2.3.

3. "There are standard object relationship queries." We alluded to some of these in discussing the Bat database.

4. "Uncertainty must be preserved." Uncertainty output of location algorithms can often be computed as a function of the inputs [HB01b]. As we discussed in Section 5.2, applications are concerned with the level of uncertainty in a reported location, so uncertainty should be preserved through all levels of abstraction. Bulusu makes a similar observation and extends it to say that quality of service trade-offs (*e.g.*, update lag vs. energy consumption) must be exposed to the application level [BEH01].

5. "Applications are usually concerned with activities." This seems the most controversial of Hightower's conclusions. The reasoning is that applications capture context to draw conclusions about user activities.

From these principles, Hightower *et al.* have begun designing the seven layer *Location Stack*. The Location Stack is meant as an analog to the OSI networking stack. It is meant to be a common paradigm about which to organize location systems and location-aware applications. The levels are currently identified as: sensors, measurements, fusion, arrangements, contextual fusion, activities, and intentions. While the interfaces between the layers have not yet been specified, the approach seems very promising. Particularly promising are the definition of the five lower levels that provide a framework for transforming data from a variety of sensors into locations in a variety of symbologies.

6.3 Distributed Algorithms for Multilateration

As mentioned in Section 5, there are several open questions about the AHLoS system. First, it would seem worthwhile to prove Claims 1 and 2. A second challenge is to find a distributed algorithm that completely characterizes the cases when collaborative multilateration can be applied. Third, having identified such a case, it is a separate challenge to find an efficient distributed implementation of collaborative multilateration. In particular it would be nice to minimize network traffic and local computation while maintaining a fair expenditure of resources. Fourth, finding an appropriate group membership algorithm appropriate for the extremely dynamic and the low power environments of ad hoc networks will be difficult. Perhaps, an alternative method of finding a consistent cut is possible.

Finally, in the absence of node movement, it is not at all clear that there any reason to iterate collaborative multilateration as the authors do. In the atomic case, it is clear that a newly located node may become the fourth beacon for some other unknown. However, it should be possible to prove that once a collaborative subgraph has been identified and solved, it cannot be sufficiently connected to any other subgraph that will be collaborative (even after the addition of the new beacons).

6.4 Maximizing Beacon Rates

One simple issue not studied in the literature surveyed is the tradeoff of Cricket beacon density and update rate. Clearly, if the there is only one Cricket beacon, it can be set to broadcast every 20ms (limited by ultrasound reverberations). As the number of beacons in a room grows, the probability of interfering broadcasts grows as well. The *beacon density problem* can be stated as an optimization problem.

Let $G = (N, E)$ be a graph of beacons where there is a node for each beacon and an edge between beacons whose broadcasts may interfere. In each discrete time step, each node flips a coin with some probability distribution D. If the coin comes up heads, the node broadcasts. If two neighboring nodes broadcast in the same time step, the broadcasts are said to interfere. The beacon density problem is to find the probability distribution D that gives the highest expected number of non-interfering broadcasts across G in any time step.

Clearly there are many interesting variations on this problem. For example, we can assume G is random or that G has maximum or minimum degree. Alternatively, the problem can be reformulated so that broadcasts may be considered to occur in a continuous time domain. Further, we may wish to prove fairness properties about the distribution.

6.5 Bootstrapping

Providing initial positions (whether for receiver antennae or beacons) in systems such as Active Bats, Cricket Compass, or even AHLoS, is a large percentage of the installation overhead. Given a sufficiently connected system, locating a few initial nodes (the number depending on the degrees of freedom) is sufficient to determine the location of the entire system. For example, if a stable AHLoS network has minimum degree of three and at least three beacons, one pass of collaborative multilateration can compute the location of every node in the system.

Unfortunately, such a high degree of connectivity is not usual. However, is seems quite possible to temporarily boost the local degree of connectivity by placing a number of mobile units as a bridge between a section of network with known locations and a section without. For example, consider a hallway that has been calibrated and a room that is newly outfitted with an array of four Cricket Compass beacons at unknown locations. None of the four room beacons may have line of sight on four hallway beacons. However, we can place a number of mobile units in or near the doorway with line of sight on both the hallway and the room. The units can first locate themselves using the hallway beacons and then be used to locate the beacons in the room. This sort of creeping localization builds on the iterative multilateration method of Savvides *et al.* Of course, as previously mentioned, bounding the error of such a system would be a challenge.

It would be particularly useful to bootstrap an indoor location system from a set of reference points with known locations in a geographic location system such as GPS.

7 Conclusions

We have identified common techniques and technologies for estimating the location of pervasive computing elements indoors. These techniques include time of flight and signal strength methods for ranging and angulation. The combination of simultaneous ultrasound and RF broadcasts combined by multilateration seems to be the single most promising technique for estimating location, capable of 10cm, 5° precision. We evaluated the Ad Hoc Location System, RADAR, Cricket, Cricket Compass, and Active Bats systems to identify commonalities and contrasts. Some of the largest contrasts come in the designers approach to centralization or distribution. Active Bats is a completely centralized system, Cricket a completely decentralized one, and AHLoS a cooperative distributed system. We identified a number of weaknesses in all the current prototypes including algorithmic shortcomings, lack of robustness to systematic error, scaling effects, and installation overhead. Finally we identified research several opportunities. Greater emphasis on software support is needed to allow sensor fusion to overcome scaling and robustness problems. Further algorithmic research is needed to find and analyze distributed localization and beaconing algorithms. Finally, the is a large opportunity to lower installation costs by investigating bootstrapping techniques.

Acknowledgements

I thank Chris Luhrs for his insights into the AHLoS collaborative multilateration algorithm. He was able to identify both a reasonable definition of participation and a counterexample to show that participation is not complete. I am also happy to acknowledge the guidance of Steve Garland and Dorothy Curtis with respect to the relevant issues in location systems.

References

[ACH$^+$01] Mike Addlesee, Rupert Curwen, Steve Hodges, Joe Newman, Pete Steggles, Andy Ward, and Andy Hopper. Implementng a sentient computing system. *IEEE Computer*, 34(8):2–8, August 2001.

[BEH01] Nirupama Bulusu, Deborah Estrin, and John Heidemann. Tradeoffs in location support systems: The case for quality-expressive location models for applications. In *Proceedings of the Ubicomp Workshop on Location Modeling*, pages 7–12, Atlanta, GA, September 2001.

[BHE00] Nirupama Bulusu, John Heidemann, and Deborah Estrin. GPS-less low cost outdoor localization for very small devices. *IEEE Personal Communications, special issue on "Smart Spaces and Environments"*, 7(5):28–34, October 2000.

[BHE01] Nirupama Bulusu, John Heidemann, and Deborah Estrin. Adaptive beacon placement. In *Proceedings of the 21st International Conference on Distributed Computing Systems*, pages 489–498, Phoenix, AZ, April 2001.

[BP00a] Paramvir Bahl and Venkata N. Padmanabhan. Enhancements to the RADAR user location and tracking system. MSR-TR 2000-12, Microsoft Research, Redmond, WA, February 2000.

[BP00b] Paramvir Bahl and Venkata N. Padmanabhan. RADAR: An in-building RF-based user location and tracking system. In *Proceedings of IEEE Infocom 2000*, volume 2, pages 775–784, March 2000.

[GDL$^+$01] Robert Grimm, Janet Davis, Eric Lemar, Adam MacBeth, Steven Swanson, Steven Gribble, Tom Anderson, Brian Bershad, Gaetano Borriello, , and David Wetherall. Programming for pervasive computing environments. UW-CSE 01-06-01, Department of Computer Science and Engineering, University of Washington, Seattle, WA, June 2001.

[GE01] Lewis Girod and Deborah Estrin. Robust range estimation using acoustic and multimodal sensing. In *Proceedings of IEEE/RSJ International Conference on Intellegent Robots and Systems (IROS)*, pages 28–34, October 2001.

[Gir00] Lewis Girod. Development and characterization of an acoustic rangefinder. USC-CS 00-728, Information Sciences Institute, University of Southern California, Marina del Rey, CA, April 2000.

[HB93] Andy Harter and Frazer Bennett. Low bandwidth infra-red networks and protocols for mobile communicating devices. ORL 93.5, Olivetti Reaserch, Cambridge, UK, October 1993.

[HB01a] Jeffrey Hightower and Gaetano Borriello. Location systems for ubiquitous computing. *IEEE Computer*, 34(8):57–66, August 2001.

[HB01b] Jeffrey Hightower and Gaetano Borriello. Real-time error in location modeling for ubiquitous computing. In Michael Beigel, Phil Gray, and Daniel Salber, editors, *Location Modeling for Ubiquitous Computing - Ubicomp 2001 Workshop Proceedings*, pages 21–27, Atlanta, GA, September 2001.

[HB01c] Jeffrey Hightower and Gaetano Borriello. A survey and taxonomy of location sensing systems for ubiquitous computing. UW CSE 01-08-03, Department of Computer Science and Engineering, University of Washington, Seattle, WA, August 2001.

[HBB02] Jeffrey Hightower, Barry Brumitt, and Gaetano Borriello. The Location Stack: A layered model for location in ubiquitous computing. In *Proceedings of the*

4th IEEE Workshop on Mobile Computing Systems and Applications (WMCSA 2002), pages 22–28, Callicoon, NY, June 2002. IEEE Computer Society Press.

[HHS+99] Andy Harter, Andy Hopper, Pete Steggles, Andy Ward, and Paul Webster. The anatomy of a context-aware application. In *Proceedings of the 5th Annual ACM International Conference on Mobile Computing and Networking (MobiCom 1999)*, pages 59–68, August 1999.

[HVBW01] Jeffrey Hightower, Chris Vakili, Gaetano Borriello, and Roy Want. Design and calibration of the SpotON ad-hoc location sensing system. `http://portolano.cs.washington.edu/projects/spoton/`, August 2001.

[KH00] Hiroaki Koshima and Joseph Hoshen. Personal locator services emerge. *IEEE Spectrum*, 37(2):41–48, February 2000.

[KS97] Dana Kirsch and Thad Starner. The locust swarm: An environmentally-powered, networkless location and messaging system. In *Proceedings of the 1st IEEE International Symposium on Wearable Computers*, pages 169–170, October 1997.

[Miu02] Allen Ka Lun Miu. Design and implementation of an indoor mobile navigation system. Master's thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA, January 2002.

[OA00] Robert J. Orr and Gregory D. Abowd. The smart floor: A mechanism for natural user identification and tracking. In *Comanion Proceedings of the 2000 Conference on Human Factors in Computing Systems (CHI 2000)*, volume 2, pages 1–2, April 2000. Short paper.

[Ome02] Mensur Omerbashich. Integrated INS/GPS navigation from a popular perspective. *Journal of Air Transportation (JAT)*, 7(1):103–118, February/March 2002. Online tutorial version `einstein.gge.unb.ca/tutorial/precision_navigation.htm`.

[PCB00] Nissanka B. Priyantha, Anit Chakaborty, and Hari Balakrishnan. The Cricket location-support system. In *Proceedings of the 6th Annual ACM International Conference on Mobile Computing and Networking (MobiCcom 2000)*, pages 32–43, August 2000.

[PMBT01] Nissanka B. Priyantha, Allen K. L. Mui, Hari Balakrishnan, and Seth J. Teller. The Cricket compass for context-aware mobile applications. In *Proceedings of the 7th Annual ACM International Conference on Mobile Computing and Networking (MobiCom 2001)*, pages 1–14, July 2001.

[RCB+02] Sanjay Raman, Dwaine Clarke, Matt Burnside, Srinivas Devadas, and Ronald Rivest. Access-controlled resource discovery for pervasive networks. Csg, Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, MA, June 2002.

[SHS01] Andreas Savvides, Chih-Chieh Han, and Mani B. Srivastava. Dynamic fine-grained localization in ad-hoc wireless sensor networks. In *Proceedings of the 7th Annual ACM International Conference on Mobile Computing and Networking (MobiCom 2001)*, pages 166–179, July 2001. Also NESL TR TM-UCLA-NESL-2001-01-001, January 2001.

[SR02] Scott Y. Seidel and Theodore S. Rappaport. 914 MHz path loss prediction models of indoor wireless communications in multifloored buildings. *IEEE Transactions on Antennas and Propagation*, 40(2):207–217, February 2002.

[Van01] Petr Vaníček. Geodesy. In Robert A. Meyers, editor, *Encyclopedia of Physical Science and Technology*, volume 6. Academic Press, New York, third edition, October 2001. Online tutorial version `einstein.gge.unb.ca/tutorial/tutorial.htm`.

[WADP97] Christopher R. Wren, Ali Azarbayejani, Trevor Darrell, and Alex Pentland. Pfinder: Real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):780–785, July 1997.

[War98] Andrew Martin Robert Ward. *Sensor-driven Computing.* PhD thesis, Corpus Christi College, University of Cambridge, Cambridge, UK, August 1998.

[WHFG92] Roy Want, Andy Hopper, Veronica Falcao, and Jon Gibbons. The active badge location system. *ACM Transactions on Office Information Systems (TOIS)*, 10(1):91–102, January 1992.

[WJH97] Andy Ward, Alan Jones, and Andy Harper. A new location technique for the active office. *IEEE Personal Communications*, 4(5):42–47, October 1997.

[WL98] Jay Werb and Collin Lanzl. Designing a positioning system for finding things and people indoors. *IEEE Spectrum*, 35(9):71–78, September 1998.